Manual/PartI/Your First Animation in 30 plus 30 Minutes Part II

From BlenderWiki

< Manual | PartI

User Manual: Contents | Guidelines | Blender Version **2.40**

Contents

- 1 Your First Animation in 30 plus 30 Minutes Part II
  - 1.1 Rigging
  - 1.2 Skinning
    - 1.2.1 Vertex groups
  - 1.3 Posing
    - 1.3.1 Original position
    - 1.3.2 Inverse Kinematics
    - 1.3.3 Forward Kinematics
  - 1.4 Gus walks!

Your First Animation in 30 plus 30 Minutes Part II

If we were going for a still picture, our work up to this point would be enough, but we want Gus to move! The next step is to give him a skeleton, or Armature, which will move him. This is called the fine art of rigging. Gus will have a very simple rigging: four limbs (two arms and two legs) and a few joints (no elbows, only knees), but no feet or hands.

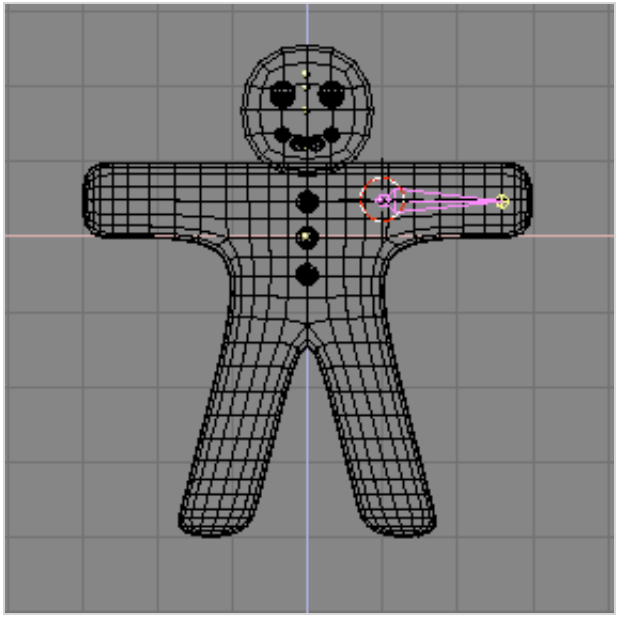
Rigging

To add the rigging:

Set the cursor where the shoulder will be, press

Set the cursor where the shoulder will be, press **SPACEKEY**>>Add>>Armature. A rhomboidal object will appear, a bone of the armature system, stretching from cursor to mouse pointer. You are automatically placed in `Edit mode`.

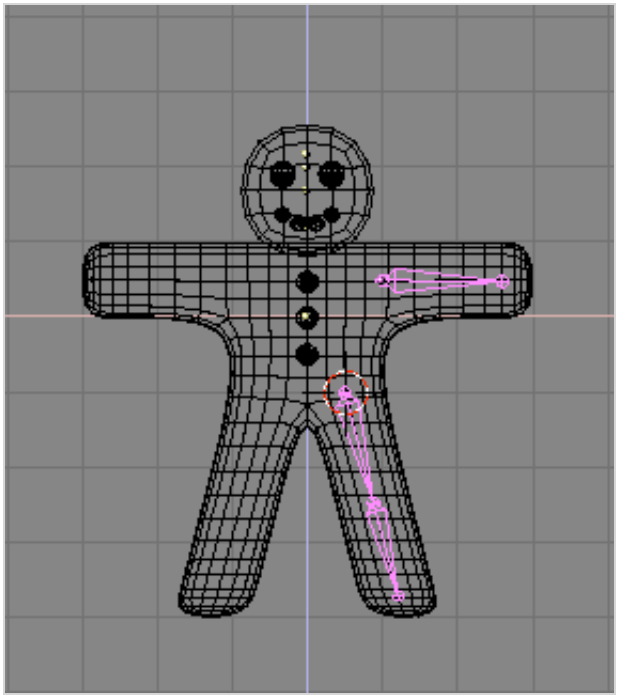
- Place the other end of the armature in Gus's hand by grabbing (**GKEY**) and moving, (*Adding the first bone, an elbowless arm*). We don't need any other bones right now, so press **ESCKEY** to exit. You should now have one bone running from the shoulder to the hand area.



Adding the first bone, an elbowless arm.

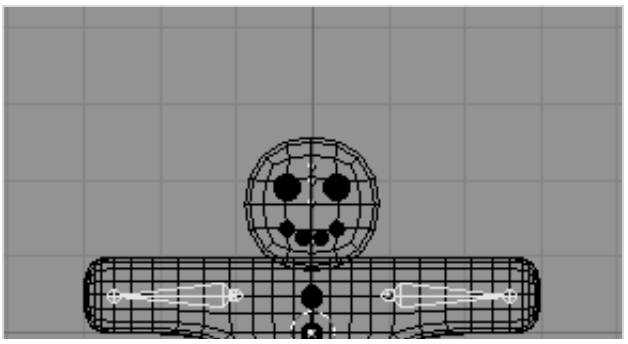
- Stay in `Edit mode`, then move the cursor to where the hip joint will be and add a new bone (**SPACE**>>Add>>Bones) or use **SHIFT-A**.
- Grab (**GKEY**) and move the yellow end of the new bone to the knee area.
- Now "chain" a new bone from the knee to the foot by **CTRL-LMB** in the area of the foot. A new "chained" bone will appear automatically linked with the knee and ending at the foot, (*Adding the second and third bones, a leg bone chain*). Another way of "chaining" the new bone would be to extrude using the (**EKEY**). This variation creates the new bone and places you in grab mode automatically. Use the **LMB** to then place the bone.

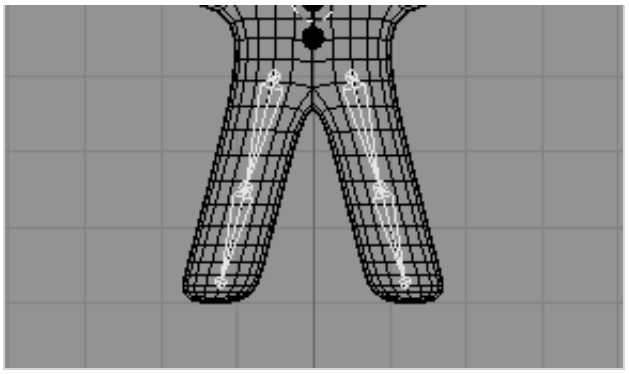
Bone position: The bones we are adding will deform Gus's body mesh. To produce a neat result, try to place the bone joints as shown in the illustrations.



Adding the second and third bones, a leg bone chain.

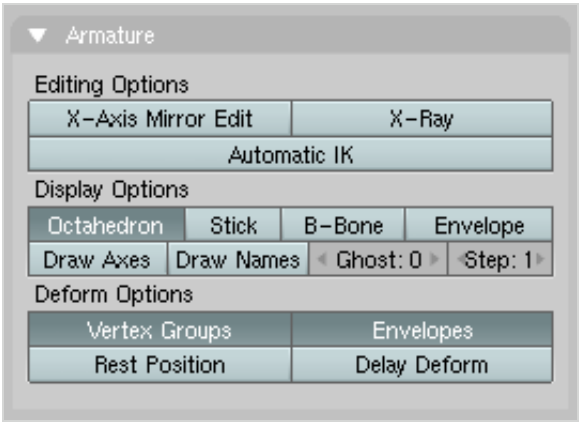
- Now place the cursor in the center and select all bones with **AKEY**. Duplicate them with **SHIFT-D** and exit grab mode with **ESCKEY** then flip them with **MKEY** relatively to the cursor and Global X axis as you did with the mesh.
- Move the duplicated bones to the left side by grabbing (**GKEY**) and moving them along the X-axis (**XKEY**). You end up with (*The complete armature after duplicating and flipping*).



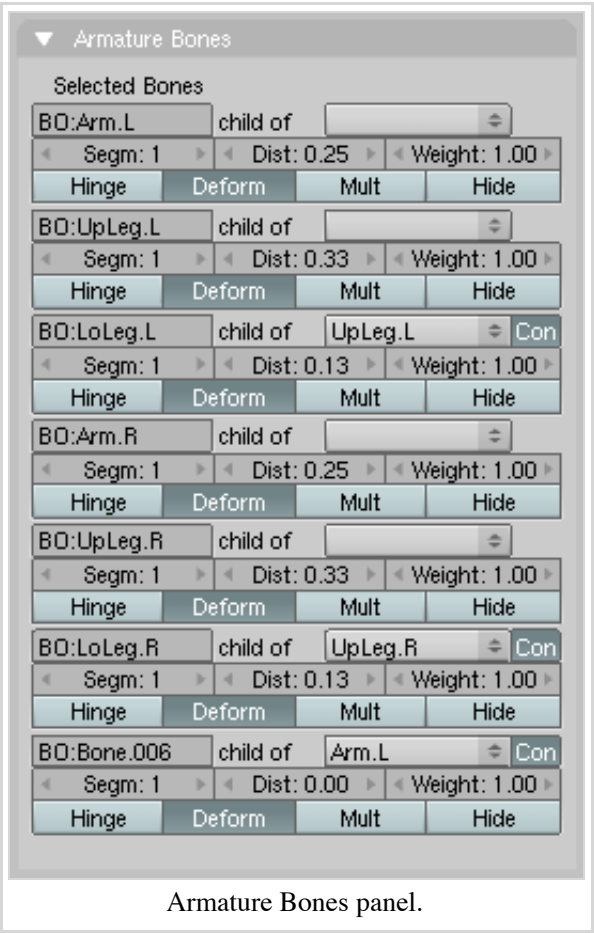


The complete armature after duplicating and flipping.

Once you've selected all of the bones (**AKEY**), the Edit Buttons window should show an `Armature` Panel and `Armature Bones` Panel which contains the Armature controls (*Armature panel* and *Armature Bones panel*).



Armature panel.



Armature Bones panel.

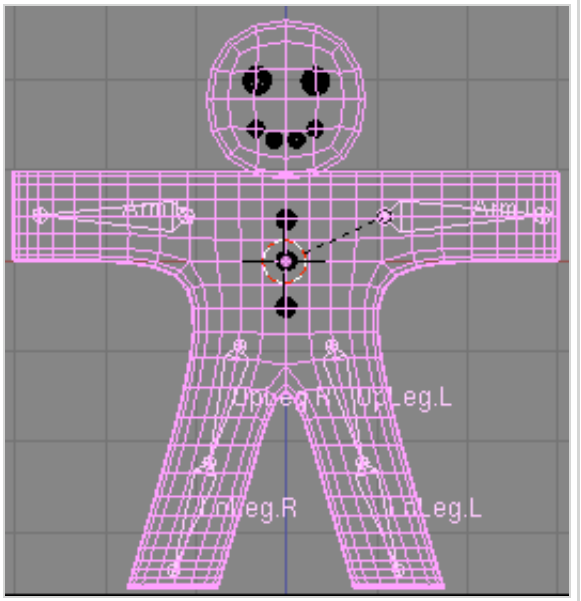
Press the `Draw Names` button to see the names of the bones, then **LMB** on the names in the Edit Button window to change them to something appropriate like Arm.R, Arm.L, UpLeg.R, LoLeg.R, UpLeg.L and LoLeg.L, see (*The Edit Buttons window for an armature*). Exit EditMode with (**TABKEY**).

Naming Bones: It is very important to name your bones with a trailing '.L' or '.R' to distinguish between left and right ones, so that the Action editor will be able to automatically flip your poses.

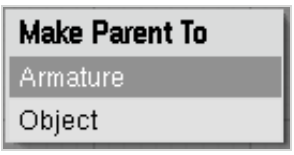
Skinning

Now we must make it such that a deformation in the armature causes a matching deformation in the body. We do this with *Skinning*, which assigns vertices to bones so that the former are subject to the latter's movements.

- Select Gus's body, then **SHIFTKEY** select the armature so that the body is magenta and the armature is light pink.
- Press **CTRL-P** to parent the body to the armature. The (*Parenting menu*) will appear. Select the `Armature` entry.



Armature parented.

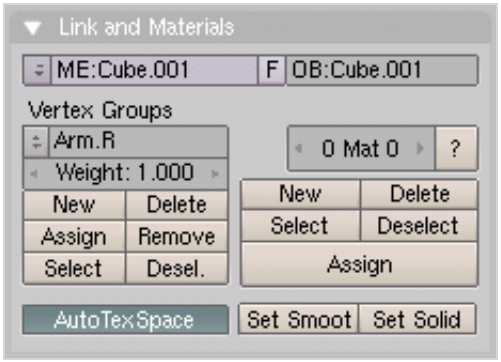


Parenting menu.

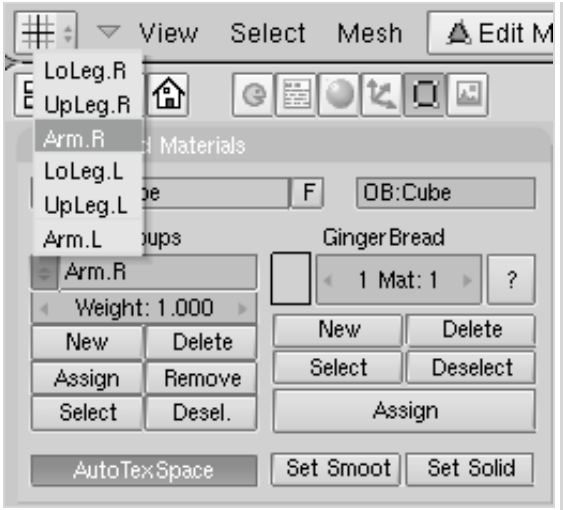
Create Vertex Group menu.

- A new menu appears, asking if you want Blender to do nothing, create empty vertex groups, or create and populate vertex groups (*Create Vertex Group menu*). The last option is considered *automatic skinning*.

- We'll use the automatic skinning option. Go ahead and select `Create From Closest Bones`. Now select just Gus's body and switch to `EditMode` (**TABKEY**). Notice in the Edit Buttons Window (**F9KEY**) the presence of the "Vertex Groups" menu and buttons in the `Link and Materials` Panel, (*The vertex groups buttons in the Edit Buttons window*).



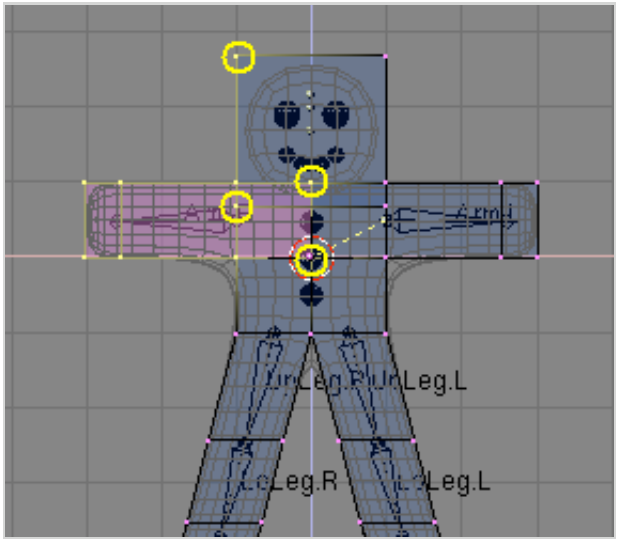
The vertex groups buttons in the Edit Buttons window



The menu with the vertex groups automatically created in the skinning process.

By pressing the `Menu Button` a menu, with all available vertex groups, pops up --six in our case. But a truly complex character, with hands and feet completely rigged, can have tens of them! See (*The menu with the vertex groups automatically created in the skinning process*). The buttons `Select` and `Deselect` show you which vertices belong to which group.

Select the Right arm group (**Arm.R**) and, with all vertices de-selected (**AKEY**, if needed) press `Select`. You should see something like (*Gus in EditMode with all the vertices of group Arm.R selected*).



Gus in EditMode with all the vertices of group Arm.R selected.

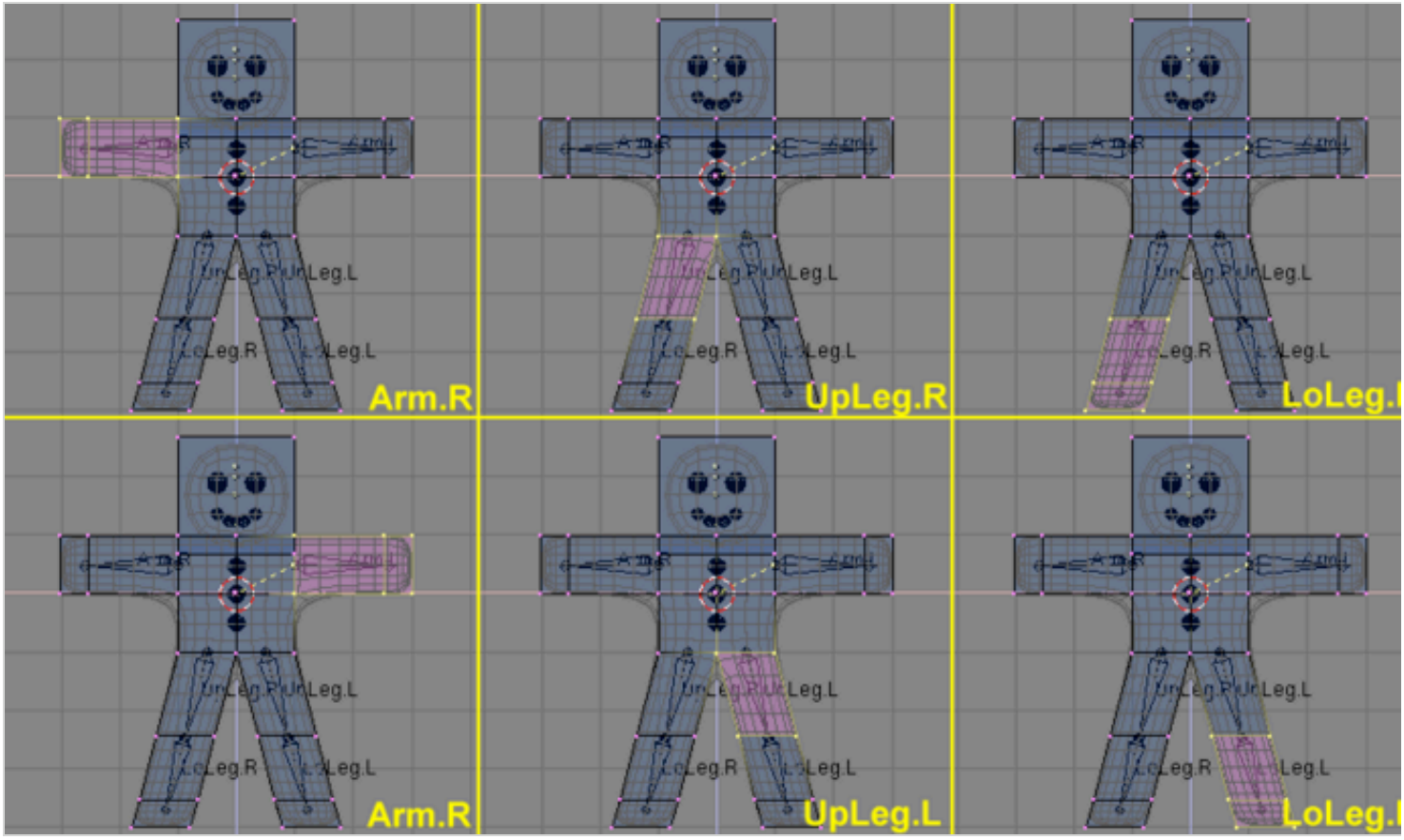
If you don't see the same thing then you problably placed the bones in just the right place such that the *auto skinning* process did a better job of matching vertices with bones. It is highly unlikely that the skinning process matched the vertices to the bones as exactly as you may expect. This requires that you begin to manually adjust the grouping as described in the following sections.

The vertices marked with yellow circles in (*Gus in EditMode with all the vertices of group Arm.R selected*) belong to the deformation group, however, they should not.

The *auto skinning* process found that they were very close to the bone so it added them to the deformation group. We don't want them in this group since some are in the head and some are in the chest, adding them to the deformation group would deform those body parts as well.

To remove them from the group, deselect all the other vertices, those which should *remain* in the group using Box selection (**BKEY**), but use **MMB**, not **LMB**, to define the box, so that all vertices within the box become deselected.

Once the 'undesired' vertices are selected, press the `Remove` button (*The vertex groups buttons in the Edit Buttons window*) to eliminate them from group (**Arm.R**). Deselect all (**AKEY**) then check another group. Check them all and be sure that they look like those in (*The six vertex groups*).



The six vertex groups.

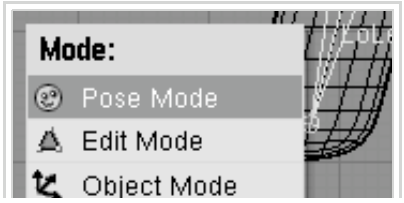
Vertex groups

Be very careful when assigning or removing vertices from vertex groups. If later on you see unexpected deformations, you might have forgotten some vertices, or placed too many in the group. You can modify your vertex groups at any time.

> **Other details:** Our deformations will affect only Gus's body, not his eyes, mouth, or buttons, which are separate objects. While this is not an issue to consider in this simple animation, it's one that must be taken into account for more complex projects, for example by parenting or otherwise joining the various parts to the body to make a single mesh. (We'll describe all of these options in detail in later Chapters).

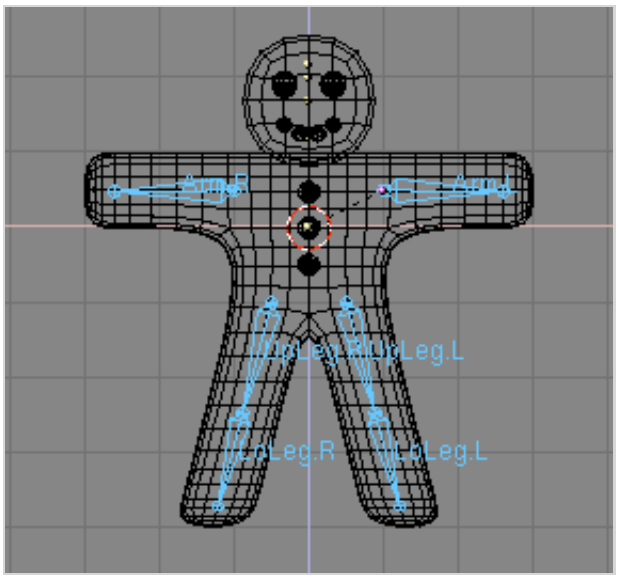
Posing

Once you have a rigged and skinned Gus you can start playing with him as if he were a doll, moving his bones and viewing the results.



- Select the armature only, then select `Pose Mode` from the "Mode" Menu (*Mode menu in the 3D Window header*). This option only appears if an armature is selected.
- The armature will turn blue. You are now in `Pose Mode`. If you now select a bone it will turn cyan, not pink, and if you move it (**GKEY**), or

rotate it (**RKEY**), the body will deform!



You are in pose mode now!

Mode menu in the 3D Window header.

Original position

Blender remembers the original position of the bones. You can set your armature back by pressing the `RestPos` button in the Armature Edit Buttons (*The Edit Buttons window for an armature*.).

Inverse Kinematics

Inverse Kinematics (`IK`) is where you actually define the position of the *last* bone in the chain, often called an "*End Effector*". All the other bones assume a algorithmic position, automatically computed by the *IK* solver, to keep the chain without gaps (i.e. `IK` will mathematically solve the chain positions for us). This allows a much easier and precise positioning of hands and feet using `IK`.

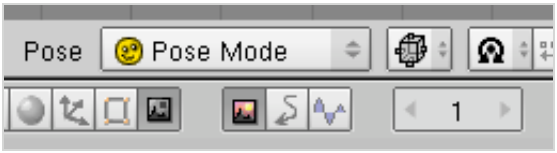
Forward Kinematics

While handling bones in `Pose Mode` notice that they act as rigid, inextensible bodies with spherical joints at the end. You can grab only the first bone of a chain and all the others will follow it. All subsequent bones in the chain cannot be grabbed and moved, you can only rotate them, so that the selected bone rotates with respect to the previous bone in the chain while all the subsequent bones of the chain follow its rotation.

This procedure, called *Forward Kinematics (`FK`)*, is easy to follow but it makes precise location of the last bone in the chain difficult.

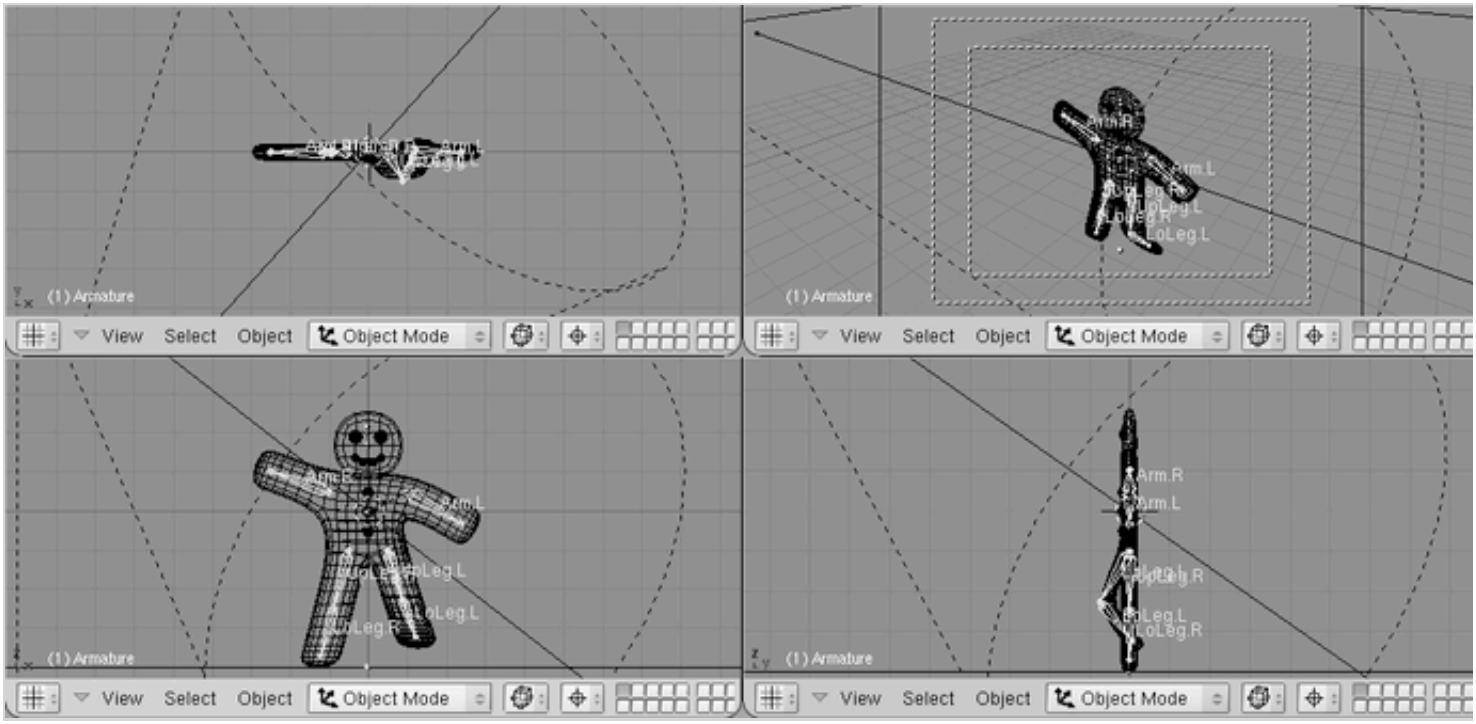
We'll make Gus walk, using `FK`, by defining four different poses relative to four different stages of a stride. Blender will do the work of creating a fluid animation.

- First, verify that you are at frame **1** of the timeline. The frame number appears in a NumButton on the right of the Buttons Window Toolbar (*The current frame Num Button in the Buttons window Toolbar*.). If it is not set to **1**, set it to **1** now.
- Now, by rotating only one bone at a time (**RKEY**), we'll raise **UpLeg.L** and bend **LoLeg.L** backwards while raising



The current frame Num Button in the Buttons window Toolbar.

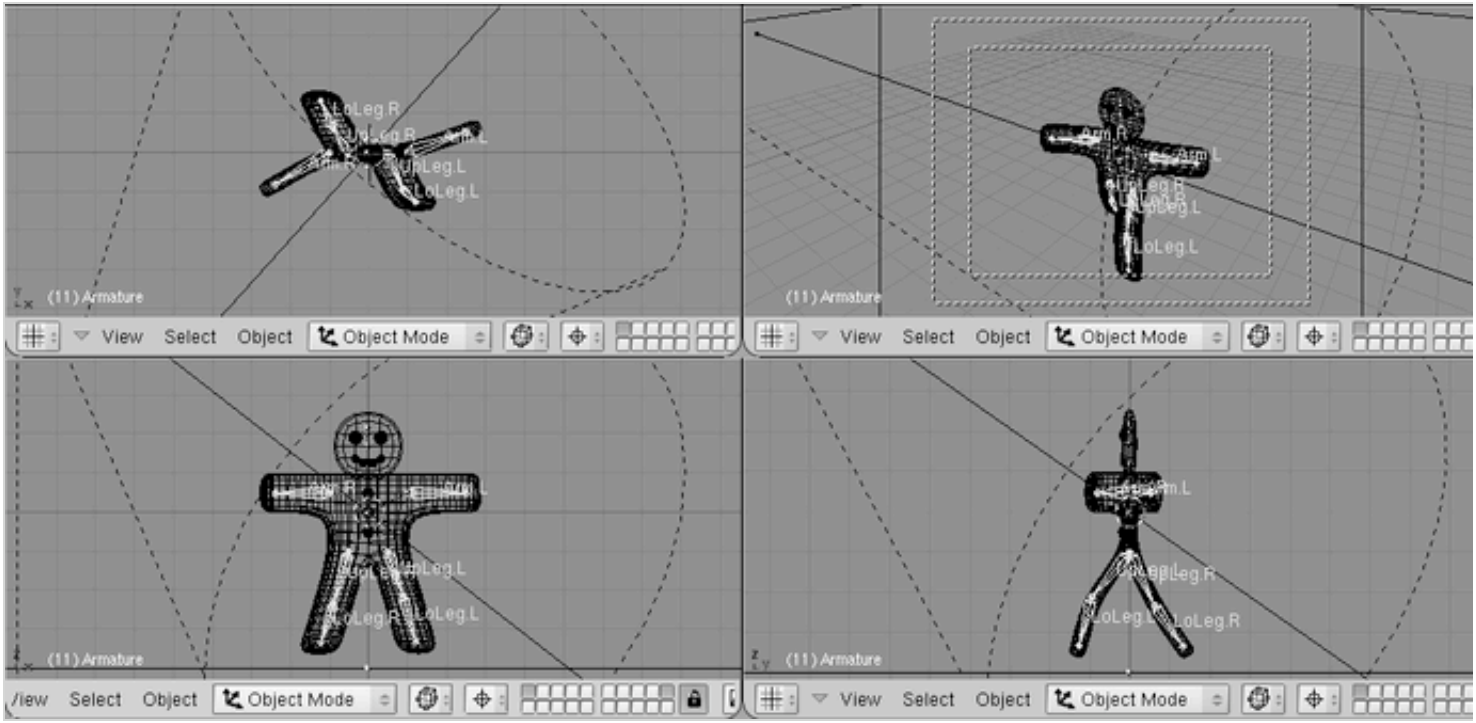
Arm.R a little and lowering **Arm.L** a little, as shown in *Our first pose*..



Our first pose.

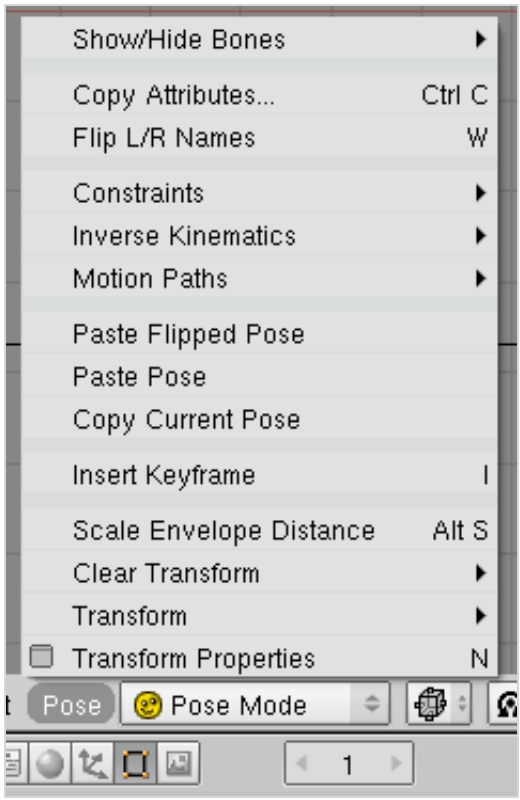
- Select all bones with **AKEY**. With the mouse pointer on the 3D Window, press **IKEY**. A menu pops up (*Storing the pose to the frame*). Select `LocRot` from this menu. This will get the position and orientation of all bones and store them as a pose at frame 1. This pose represents Gus in the middle of his stride, while moving his left leg forward and above the ground.
- Now move to frame 11 either by entering the number in the NumButton or by pressing **UPARROW**. Then move Gus to a different position, like (*Our second pose*), with his left leg forward and right leg backward, both slightly bent. Gus is walking in place!

Storing the pose to the frame.



Our second pose.

- Select all bones again and press **IKEY** to store this pose at frame 11.
- We now need a third pose at frame 21, with the right leg up, because we are in the middle of the other half of the stride. This pose is the mirror of the one we defined at frame 1. Therefore, return to frame 1 and, in the `Pose` Menu in the 3D Window header select the `Copy Current Pose` entry, see (*Pose menu*). You have now copied the current pose to the buffer.
- Go to frame 21 and paste the pose with the `Paste Flipped Pose` option in the `Pose` Menu, see (*Pose menu*). This button will paste the cut pose, exchanging the positions of bones with suffix "**.L**" with those of bones with suffix "**.R**", effectively flipping it!



Pose menu

The pose is there but it has not been stored yet! You must press **IKEY** with all bones selected.

- Now apply the same procedure to copy the pose at frame 11 to frame 31, also flipping it.
- To complete the cycle, we need to copy the pose at frame 1 *without* flipping to frame 41. Do so by copying it as usual, and by using the `Paste Pose` entry. End the sequence by storing the pose with **IKEY**.

Checking the animation: To preview your Animation, set the current frame to 1 and press **ALT-A** in the 3D window.

Gus walks!

The single step in-place is the core of a walk, and once you have defined one there are techniques to make a character walk along a complex path. But, for the purpose of our Quick Start, this single step in-place is enough.

- Change to the Rendering Buttons (**F10**) and set the animation start and end to 1 and from 250 to 40 in the `Anim` Panel respectively (*Setting the Rendering Buttons for an animation*). Because frame 41 is identical to frame 1, we only need to render frames from 1 to 40 to produce the full cycle.

Setting the Rendering Buttons for an animation.

- Select AVI Raw as the file type in `Format` Panel (*Setting the Rendering Buttons for an animation.*). While this is generally not the best choice, mainly for file size issues (as will be explained later on), it is fast and it will run on any machine, so it suits our needs. (You can also select AVI Jpeg to produce a more compact file. However, it uses lossy JPEG compression and will produce a movie that some external players might not be able to play).

Finally, press the `ANIM` button in `Anim` Panel. Remember that *all* the layers that you want to use in the animation must be shown! In our case, these are layers 1 and 10.

Stopping a Rendering: If you make a mistake, like forgetting to turn layer 10 on, you can stop the rendering process with the **ESC** key.

Our scene is pretty simple, and Blender will probably render each of the 40 images in a few seconds. Watch them as they appear.

Stills: Of course you can always render each of your animation frames as a still by selecting

the frame you wish to render and pressing the `RENDER` button.

Once the rendering is complete you should have a file named `0001_0040.avi` in a `render` subdirectory of your current directory --the one containing your `.blend` file.

You can play this file directly within Blender by pressing the {LiteralPlay}} button beneath the `ANIM` button (*Setting the Rendering Buttons for an animation*). The animation will automatically cycle. To stop it press **ESC**. We have produced only a very basic walk cycle. There is much more in Blender, as you'll soon discover!

Previous: Manual/PartI/Your_First_Animation_in_30_plus_30_Minutes_Part_I Contents *Next:* Manual/PartII

Retrieved from
"http://mediawiki.blender.org/index.php/Manual/PartI/Your_First_Animation_in_30_plus_30_Minutes_Part_II"

- This page was last modified 14:16, 21 April 2006.